

Assignment

1. Define Artificial Intelligence (AI) and provide examples of its applications:

AI refers to creating intelligent machines capable of tasks typically associated with human intelligence. It involves learning from data, adapting to new situations, and making decisions.

Applications:

- **Healthcare:** AI assists in diagnostics, personalized treatment, and drug discovery.
- **Finance:** Fraud detection, algorithmic trading, and credit scoring.
- **Natural Language Processing (NLP):** Chatbots, language translation, and sentiment analysis.
- **Robotics:** Autonomous vehicles, drones, and industrial automation.
- **Recommendation Systems:** E-commerce, content recommendations, and personalized ads.

2. Differentiate between supervised and unsupervised learning techniques in ML:

- **Supervised Learning:**
 - Uses labeled data (input-output pairs) for training.
 - Examples: Regression (predicting continuous values) and Classification (assigning data to predefined categories).
- **Unsupervised Learning:**
 - Uses unlabeled data to discover patterns and insights.
 - Examples: Clustering (grouping similar data) and Dimensionality Reduction (simplifying data representation).

3. What is Python? Discuss its main features and advantages:

- **Definition:** Python is a dynamic, high-level, free open-source, and interpreted programming language.
- **Features:**
 - **Easy to Learn:** Simple syntax and developer-friendly.
 - **Object-Oriented:** Supports classes, objects, and encapsulation.
 - **GUI Programming Support:** Modules like PyQt5 for graphical apps.
 - **Large Community Support:** Active StackOverflow community.
 - **Portable and Integrated:** Runs on various platforms and integrates with other languages.

4.What are the advantages of using Python as a programming language for AI and ML?:

- **Readability:** Clear syntax and indentation enhance code readability.
- **Rich Libraries:** Extensive libraries (e.g., NumPy, Pandas, TensorFlow) for AI/ML tasks.
- **Scalability:** Python scales well for large projects.
- **Community Support:** Active Python community and vast resources.
- **Integration:** Easily integrates with other languages.
- **Interpreted Language:** No need to manage memory or system architecture.

5.Discuss the importance of indentation in Python code:

- **Definition:** Python uses indentation to define code blocks.
- **Why?:**
 - Proper indentation ensures code readability.
 - Indentation errors lead to IndentationError.
 - It enforces a clean coding style and highlights code blocks.
- **Example:**

Python

```
if site == 'gfg':
    print('Logging on to geeksforgeeks...')
else:
    print('retype the URL.')
```

```
print('All set!')
```

6. Define a variable in Python. Provide examples of valid variable names:

- A variable stores data. Example: age = 25.
- Explain the difference between a keyword and an identifier in Python:
- **Keyword:** Reserved words (e.g., if, else, while) with predefined meanings.
- **Identifier:** User-defined names (e.g., variable names) following rules (start with a letter/underscore, no spaces).
- List the basic data types available in Python:
- **Integers:** Whole numbers (e.g., 42).
- **Floats:** Decimal numbers (e.g., 3.14).
- **Strings:** Text (e.g., "Hello, World!").
- **Booleans:** True or False.
- Describe the syntax for an if statement in Python:
- Syntax:

Python

```
if condition:
```

```
    # Code block executed if condition is True
```

7. Explain the purpose of the elif statement in Python:

- Used for multiple conditional checks after an initial if.
- Executes a different block of code if the first condition is False.
- Example:

Python

```
if x > 0:
```

```
    print('Positive')
```

```
elif x < 0:
```

```
    print('Negative')
```

```
else:
```

```
print('Zero')
```

8.List the basic data types available in Python:

- **Integers:** Whole numbers (e.g., 42).
- **Floats:** Decimal numbers (e.g., 3.14).
- **Strings:** Text (e.g., "Hello, World!").
- **Booleans:** True or False.

9.Describe the syntax for an if statement in Python:

- Syntax: **Python**

if condition:
 # Code block executed if condition is True

10.Explain the purpose of the elif statement in Python:

- Used for multiple conditional checks after an initial if.
- Executes a different block of code if the first condition is False.
- Example: **Python**

- if x > 0:
 print('Positive')
- elif x < 0:
 print('Negative')
- else:
 print('Zero')